

REMARKS/ARGUMENTS

Claims 1-20 were presented for examination and claims 1-5, 7-11, 13-17, 19 and 20 remain pending in this application. In an Official Final Office Action dated May 4, 2007, claims 1-5, 7-11, 13-17, 19 and 20 were rejected. The Applicant thanks the Examiner for his consideration and addresses the Examiner's comments concerning the claims pending in this application below.

Applicant herein amends claims 1, 4, 8, 11, and 16 and respectfully traverses the Examiner's prior rejections. Claims 2, 3, 7, 14, and 15 are presently cancelled without prejudice and no new claims are presently added. These changes are believed not to introduce new matter, and their entry is respectfully requested as the matter was previously presented in the cancelled dependent claims and places the independent claims in a better position for appeal. The claims have been amended to expedite the prosecution and issuance of the application. In making this amendment, the Applicant has not and is not narrowing the scope of the protection to which the Applicant considers the claimed invention to be entitled and does not concede, directly or by implication, that the subject matter of such claims was in fact disclosed or taught by the cited prior art. Rather, the Applicant reserves the right to pursue such protection at a later point in time and merely seeks to pursue protection for the subject matter presented in this submission.

Based on the above amendment and the following remarks, Applicant respectfully requests that the Examiner reconsider all outstanding rejections and withdraw them.

35 U.S.C. §112 Rejection of Claims

Claim 16 was rejected under 35 U.S.C. § 112, first paragraph, as failing to comply with the written description requirement. Specifically, claim 16 was rejected as based on a disclosure which is not enabling with respect to execution of object

configurations. Claim 16 is herein amended to remove the term “object configurations”.

Rejection of the Claims under 35 U.S.C. §102(b)

Claims 1-5, 7-11 and 13-15 were rejected under 35 U.S.C. §102(b) as being anticipated by Hennefeld, Julien O. *Using Turbo Pascal 4.0-6.0*, 2nd edition, Boston, MA 1992, 1989 (“Hennefeld”) and Inman, D. et al. *Qbasic Made Easy*, Berkeley, CA, 1991 (“Inman”). Applicant respectfully traverses these rejections in light of the following remarks.

MPEP §2131 provides:

“A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference.” *Verdegall Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 U.S.P.Q.2d 1051, 1053 (Fed. Cir. 1987). “The identical invention must be shown in as complete detail as contained in the claim.” *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989).

The claims as currently amended recite features lacking in the applied references. For example, independent claim 1 (and claims 11 and 16 in varying language) recites, among other things, “processing the code entry, wherein the processing includes comparing the code entry to a set of syntax and language rules for the programming language to identify errors by comparing syntax of the code entry to a set of syntax rules for the programming language to identify a syntax error, comparing the code entry to a set of language rules to identify a language rule violation by the code entry and when no errors are identified, executing the code entry, and when the comparing identifies the syntax error or the language rule violation, retrieving an error code based on the syntax error or language rule

violation;....” Neither Hennefeld nor Inman disclose at least this aspect of the invention.

Before addressing the specific claim language of claims 1, 11 and 16 and how they differ from Hennefeld and Inman it may be worthwhile discussing more generally the Applicant’s invention as described in the specification.

The present invention enables a programmer to view syntax errors and language rule violations on a line by line basis. Each single entry is compared against particular language syntax rules and either validated as complying with the syntax rules or identified as possessing a syntax error. When an error occurs the user is provided with immediate feedback enabling the user to learn from his or her mistake. Similarly each single entry is compared to language rules and when a violation of a language rule is identified the user is provided with immediate feedback.

Hennefeld and Inman both provide classic syntax and logical error debugging code which, in the case of a syntax error, conveys that an error has occurred but does so as the entire program is being compiled. Hennefeld, for example, halts the compilation of the entire program when an error is detected and then displays an error code for that first detected error. A single code entry has not been submitted to the compiler in Hennefeld. In fact, multiple code entries have been submitted and during the compilation of these code entries the process stops when the first error is detected.

Similarly neither Hennefeld nor Inman disclose the comparison of the single line of code to language rules to determine whether a language rule violation has occurred. Responsive to having no syntax errors, the present invention compares the single line of code to language rules to provide immediate feedback to the user whether the line of code, while possessing no syntax errors, nonetheless violates a language rule.

Inman appears to disclose debugging logical mistakes by enabling a user to

step through code, line by line, and observe the result register. This presumably provides the user with feedback as to how each code entry affects the variables. This process inherently assumes that no syntax errors or rule violations are present and is only useful to detect logical errors. Again, the entire code must be compiled as a whole and thereafter executed and Inman simply pauses execution of the program to see alterations in the results. This is not a display of the semantics of a single line of code.

The Applicant also disagrees with the characterization of Inman. Inman appears to teach an intermediate window that executes a QBasic command. QBasic is an integrated development environment (IDE) and interpreter for a variant of the BASIC programming language which is based on QuickBasic. Each QBasic statement is a combination of keywords (code) that the window understands. Inman discloses a process where these combination of statements are executed. It does not provide the user with any feedback as to show the effects of the executed code on variables.

As each and every element of the claimed invention is not disclosed by either Hennefeld or Inman, the Applicant deems claims 1, 13 and 16, and those claims that depend therefrom, not anticipated.

35 U.S.C. §103(a) Obviousness Rejection of Claims

Claims 8-10, 16, 17, 19 and 20 were rejected under 35 U.S.C. §103(a) as being unpatentable over Inman in view of U.S. Patent No. 6,282,701 by Wygodny et al. ("Wygodny"). Inman is recognized as failing to teach or suggest a semantic view that includes effects of execution of the coding statement. Wygodny is cited as supplying this aspect, specifically Figure 11 and accompanying text. The Applicant disagrees with this conclusion.

Figure 11 of Wygodny appears to show a trace pane showing, what is referred to in Wygodny, as a trace tree. As defined in col. 18, lines 44-57, a trace

tree "displays a hierarchical tree of the sequence of function calls and returns in the client 102. The number of lines in the trace tree is shown in the trace tree pane title bar 308. The trace tree 330 is organized in a standard tree structure and the developer 112 can click on the tree control buttons to collapse or expand the view of functions belonging to lower hierarchical levels." Figure 11 shows a hierarchical representation of a sequence of functions. It does not show a semantic effect of the execution of a single line of code as is claimed.

Claim 16 and all claims depending therefrom are deemed patentable over Inman in view of Wygodny. Reconsideration is respectfully requested


Conclusion

In view of all of the above, the claims are now believed to be allowable and the case in condition for allowance which action is respectfully requested. Should the Examiner be of the opinion that a telephone conference would expedite the prosecution of this case, the Examiner is requested to contact Applicant's attorney at the telephone number listed below.

Please charge deposit account no 50-1123 the Request for Continued Examination fee of \$790 for this submittal. No other fee is believed due, however, any fee deficiency associated with this submittal may be charged to Deposit Account No. 50-1123.

1 August, 2007

Respectfully submitted,



Michael C. Martensen, No. 46,901
Hogan & Hartson LLP
One Tabor Center
1200 17th Street, Suite 1500
Denver, Colorado 80202
(719) 448-5910 Tel
(303) 899-7333 Fax